

Optimization Approach using Case Based Reasoning and Evolution Strategy

Danny K.Y. Sin¹ & James N.K. Liu²

¹Operations Division, Hong Kong Mass Transit Railway Corporation Limited, Hong Kong
Email : dannys@mtrcorp.com

²Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong
Email : csnkliu@comp.polyu.edu.hk

Abstract

This paper outlines a comparative investigation on optimization using the integration of case based reasoning and neural network with the adoption of fuzzy clustering. In this integrated model, case based reasoning is applied to build the case profiles from various forms of data set that are used as inputs of network training through neural network and case based reasoning. Machine maintenance planning process is used as an illustrated experiment. It utilizes specific expert's knowledge, which is transformed into case profiles and fuzzy membership functions through certain control rules. Fuzzy clustering is applied to the data analysis and adaptive gradient learning algorithms with various network architectures are studied. Previous results (Liu and Sin 1999) indicate that the algorithm by combining case based reasoning and time lagged recurrent network can generate better training result. Experimental results show that the integration of data clustering technique as well as learning algorithm of case based reasoning (CBR) and neural network (NN) has achieved even further improvement of 246% against the result in (Liu and Sin 1999). It is also noticed that in some particular situations, CBR provides better result than NN.

Keywords: Case-based reasoning, neural network, optimization

1. Introduction

As an illustrated experiment of our proposed model for optimization, maintenance scheduling of the Hong Kong Mass Transit Railway Corporation Limited (MTRCL) is applied. The Operations Division of Hong Kong MTRCL is required to maintain various system operations for carrying 2.2 million patronage everyday. MTR operational systems are designed and installed at a wide range of stages. They are integrated together by combining with a variety of technologies. This increases the complexity of daily system operations as well as systems maintenance. A certain amount of data of the automatic fare collection (AFC) system is selected for this study, which includes entry gates, exit gates, reversible gates, ticket issuing machines and add value machines. To maximize the serviceability of railway systems by reducing disturbances to railway operation, we need to develop an enhanced maintenance scheduling system which helps decide whether to perform maintenance

before the machine fails or to defer the planned maintenance that is deemed to be unnecessary. Liu and Sin (1997) proposed a fuzzy-neural model which takes into consideration a list of factors that could affect machine performance. A variety of factors were studied and evaluated, the model reported more than 20% improvement against the traditional logistic regression analysis for maintenance schedule. As a further enhancement of the system in (Liu and Sin 1997), we introduce other learning algorithm by integrating the case-based reasoning with neural networks in (Liu and Sin 1999; Liu and Sin 2000).

The paper presents a framework to perform knowledge discovery by means of case based reasoning to form a set of case profiles through rules discovery and validation processes. The outputs are then trained separately by passing through an integrated model of neural network (NN) and case based reasoning (CBR). The next section gives the background of CBR and NN. Section 3 describes the data analysis, learning algorithms and adaptive networks. Section 4 gives the experimental results and the last section discusses the integration issues and conclusion.

2. Background

The advantages of combining multiple techniques to yield synergism for discovery and prediction have been widely recognized (Han et al. 1996). An example lies in the call for a juxtaposition of spectral analysis and temporal regression for studies in the social sciences. Despite the recognized need for integration, however, the calls for a unified approach to learning and prediction have gone largely unanswered to date (Kim and Novick 1993). A versatile approach to self-organization lies in neural networks (Golding and Rosenbloom 1991). Neural nets are characterized by their learning capability, the ability to improve performance over time. A closely related feature is that of generalization, relating to the recognition of new objects which are similar but not identical to previous ones. An additional characteristic relates to graceful degradation: The network fails gradually rather than catastrophically when it suffers partial damage.

To date, however, artificial neural networks have been subject to a major limitation: Protracted training periods. Hundreds or thousands of trials are usually required for satisfactory performance in various tasks. The time and effort required for training have hindered their widespread applications to practical domain (Kim and Novick 1993). To fully exploit the promise of neural nets by emulating the real-time responsiveness of biological systems, the training time must be reduced dramatically, by several orders of magnitude. The performance improvement of such magnitude will unlikely be materialized from a simple tweaking of algorithms or their parameters. A more drastic re-evaluation and improvement in technique are needed.

Certain approaches to the speed-up of learning show some promise. One of these relates to the use of declarative knowledge during the training procedure. More specifically, prior knowledge encoded in propositional logic may be used to define an initial structure for a neural net. In addition to the slow rate of learning, another shortcoming of the neural approach lies in the implicit nature of the learned skill. In particular, a neural network may yield the correct response to a query but it cannot explain the result or justify its reasoning. The human mind can learn a concept through a single example and, further, convey the learned message to an observer. An obvious distinction between biological and artificial networks is found in the additional capacity for symbolic reasoning in the natural kind.

The use of explicit knowledge allows for some explanation and justification for the benefit of other entities, including an interested human observer. Examples of such high-level representation, also called the *knowledge level*, lie with declarative logic such as production rules. A sophisticated learning system should provide for the fusion of both implicit and explicit methods of knowledge representation. In this way, it can build on the respective advantages of disparate techniques.

3. Methodology

3.1 Data Similarity

Similarity measures play a central role in case-based reasoning (Burkhard and Richter 2001; Dubitzky and Azuaje 2001; Plaza et al. 1997). Most case-based systems represent cases by features and employ a similarity function to measure the similarities between new and prior cases. The similarity information is classified into two kinds: one is called qualitative similarity information which represents the similarities between cases. The other is called a relative similarity information which represents the relation between similarities of two case pairs both including the same case, i.e. whether case x is more similar to case y than to case z , or not.

1) Coding weights vector (Ishii and Wang 1996)

To find a weights vector w which satisfies *Relative Similarity Condition* (RSC) :

for each chosen set of three cases (x,y,z) , if x is more similar to y than to z then $s_w(x,y) \geq s_w(x,z)$, otherwise $s_w(x,y) < s_w(x,z)$.

A weights vector is encoded by a string of real numbers where the i^{th} real number is the weight assigned to the i^{th} feature. An example of an individual is shown in the following :

feature	f_1	f_2	f_3	f_4
weight	0.25	0.30	0.10	0.35

2) Evaluation

The fitness value of an individual is assigned to be the satisfaction degree of relative similarity information, i.e. N_{RSC} / N , where N_{RSC} is the number of the sets of three cases which satisfy RSC using weights vector w represented by the individual, and N is the number of chosen sets of the three cases. The fitness value of a population is the maximum fitness value of individuals in the population.

3) Selection

All individuals in the population are sorted by their fitness values and the first individual is the best. Crossover process is arranged to the i^{th} individual and the $(i+1)^{\text{th}}$ individual to generate $[(p+1)/2]$ individual of the next population, where $i = 1, 2, \dots, [(p+1)/2]$ and p is the population size. If the fitness value of the offspring is less than that of i^{th} parent, then adding the parent to the next population instead of the offspring. Crossover process is also arranged to the j^{th} individual and the $(p+1-j)^{\text{th}}$ individual to generate $[p/2]$ individuals of the next population, where $j = 1, 2, \dots, [p/2]$, then the average fitness of individuals in the next population could be increased.

4) Crossover

Two selected individuals are called parent-A and parent-B and assume the fitness value of parent-A is higher or equal to that of parent-B. Every crossover results in one offspring by the following steps:

- For a gene w_a in parent-A and the gene w_b at the same position as w_a in parent-B, if $w_a = w_b$ then set w_a to the same position in the offspring.
- Let the number of unset genes in the offspring be m .
 - if $m < 4$ then for every unset position in the offspring, set the average value of genes at the same position in parents to the offspring.
 - if $m \geq 4$ then: select $[m+1]/2$ genes from parent-A randomly and set them to the offspring. Let the sum of these genes and the genes set by step 1 be $sum1$ and then, let the number and sum of genes in parent-B at the unset position be n' and $sum2$ respectively. For any unset position in the

offspring, if $sum2 = 0$ then set $(1-sum1)/n'$ to the position else set $g(1-sum1)/sum2$ where g is the gene in parent-B at the unset position, so that the sum of all genes in the offspring is 1.

5) Mutation

Random exchanging a gene with another different one, or replacing a gene pair (g_1, g_2) with a new one (g'_1, g'_2) where $g_1 + g_2 = g'_1 + g'_2$ for keeping the sum of weights no change.

After crossover and mutation, the differentiation of the same individuals will be carried out for a variety of individuals in the population. Mutation will operate on the individual which is the same as each other until it is different from all other individuals.

3.2 Case-based Reasoning

Case based reasoning (CBR) is a problem solving technique that is different from other AI approaches. Instead of relying solely on general knowledge of a problem domain, CBR is able to utilize the specific knowledge of previously experienced by finding a similar past case and reusing it in the new problem situation (Aha et al. 2000; Azuaje et al. 2000; Smyth 1999). CBR is also an approach to incremental, sustained learning, since a new experience is retained each time a problem has been solved, making it immediately available for future problems (Baluja 1994; Munoz-Avila et al. 1999).

The cases used by CBR can be obtained from human experts, as a form of prior knowledge (i.e. implicit analogical knowledge), as well as by induction from raw data. Hence, an integrated system combines theory and experience. In particular, it can confirm, refine and even refute its taught knowledge in the light of new evidence. Mechanisms must be implemented to handle exceptions, conflicting rules and generally contradictions between prior knowledge (the current case base) and induction (the current evidence). In traditional CBR systems, there is no automatic provision for such arrangement. It is left to the experts to design a consistent case base, or to the user to pick among possibly conflicting alternatives.

3.3 CBR Techniques

The CBR paradigm (Aamodt and Plaza 1994; Job et al. 1999; Smyth 1999) covers a range of different methods for organizing, retrieving, utilizing and indexing the knowledge retained in past cases. Cases may be kept as concrete experiences, or a set of similar cases may form a generalized case. Cases may be stored as separate knowledge units, or split up into sub-units and distributed within the knowledge structure. Cases may be indexed by a prefixed or open vocabulary, and within a flat or hierarchical index structure. The solution from a previous case may be directly applied to the new problem, or modified according to differences between the two cases.

A general CBR cycle has the following processes (Aamodt and Plaza 1994; Aha 1998):

- a) RETRIEVE the most similar case or cases
- b) REUSE the information and knowledge in that case to solve the problem
- c) REVISE the proposed solution
- d) RETAIN the parts of this experience likely to be useful for future problem solving

The following describes the CBR cycle (Watson and Marir 1994):

1) Case Representation

Typically a case consists of:

- the problem that describes the state when the case occurred,
- the solution which states the derived solution to that problem, and
- the outcome which describes the state after the case has occurred.

Cases that comprise problems and their solutions can be used to derive solutions and evaluate the outcomes for the new problems.

2) Indexing

Case indexing involves assigning indices to cases to facilitate their retrieval. Indices should be predictive, address the purposes of the case, be abstract enough to allow for widening the use of case and be concrete enough to be recognized in future.

3) Storage

Case storage is an important aspect in designing efficient CBR systems. The case-base should be organized into a manageable structure that supports efficient search and retrieval methods. A balance has to be found between storing methods that preserve the semantic richness of cases and their indices and methods that simplify the access and retrieval of relevant cases.

4) Retrieval

A retrieval algorithm using the indices in the case-memory should retrieve the most similar cases to the current problem. The retrieval algorithm relies on the indices and the organization of the memory to direct the search to potentially useful cases. Several algorithms have been implemented to retrieve appropriate cases with serial search, hierarchical search and simulated parallel search.

CBR will be ready for large-scale problems only when retrieval algorithms are efficient at handling thousands of cases. Well known methods for case retrieval are: Nearest Neighbour, Induction, Knowledge Guided Induction and Template Retrieval. These methods can be used alone or combined into hybrid retrieval strategies (Watson and Marir 1994).

a) Nearest Neighbour

This approach involves the assessment of similarity between stored cases and the new input case, based on matching a weighted sum of features. The limitation of this approach includes problems in converging on the correct solution and retrieval times. This approach is more effective when the case base is relatively small. A typical algorithm for calculating nearest neighbour matching is listed in following formula (Watson and Marir 1994):

$$\sum_{i=1}^n w_i \times \text{sim}(f_i^I, f_i^R) / \sum_{i=1}^n w_i$$

where w is the important weighting of a feature, sim is the similarity function, and f^I and f^R are the values for feature i in the input and retrieved cases respectively.

b) Adaptation

Once a matching case is retrieved, a CBR system should adapt the solution stored in the retrieved case to the needs of the current case. Adaptation looks for prominent differences between the retrieved case and the current case and then applies formulae or rules that take those differences into account when suggesting a solution.

Several techniques have been used in CBR for adaptation (Watson and Marir 1994): Null Adaptation, Parameter Adjustment, Abstraction and Re-specialization, Critic-based Adaptation, Re-instantiation, Derivational Reply, Model-guided Repair and Case-based Substitution.

3.4 Neural Networks

The most common neural network methodology employs the backpropagation algorithm. This approach involves a layered, feedforward network structure with fully interconnected nodes from one layer to the next. The learning technique involves backward propagation of errors to aid in updating internode weights. Motivated by biological systems, artificial neural networks have learning capabilities which can be applied to the task of prediction. The general architecture of neural network model is shown in Figure 1.

In a previous study, we considered time series prediction using Time Lagged Recurrent Network (TLRN) (Aamodt and Plaza 1994). TLRN with the memory layer confined to the input can be thought of input preprocessor. The information is represented across time instead of simply across the static input patterns. The most studied TLRN network is the gamma model (Baluja 1994). It is characterized by a memory structure that is a cascade of leaky integrators. The simulator was used for the test offering a choice of memory structures. The Laguarre

memory is an improvement over the gamma memory since it trains faster.

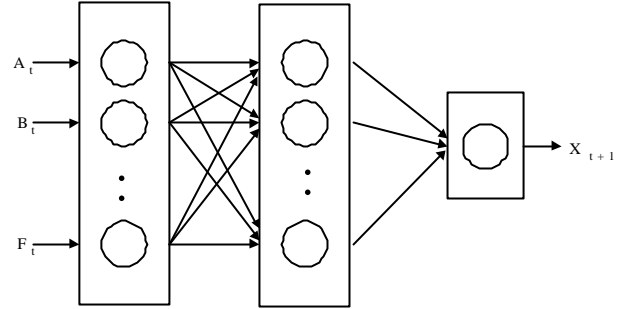


Figure 1 General architecture of the neural network model, where A_t, B_t, \dots, F_t are inputs and x_{t+1} is the trained output.

3.5 Fuzzy Rules - Linguistic Term

A first-order fuzzy association rule can be defined as a fuzzy association rule involving one linguistic term in its antecedent, a second-order fuzzy association rule can be defined as a fuzzy association rule involving two linguistic terms in its antecedent, and so on (Au and Chan 1999). Given a set of records, D , each of which consists of a set of attributes $J = \{I_1, I_2, \dots, I_n\}$, where I_1, I_2, \dots, I_n can be quantitative or categorical. For any record, $d \in D$, $d[I_v]$ denotes the value i_v in d for attribute $I_v \in J$. For any quantitative attribute, $I_v \in J$, let $\text{dom}(I_v) = [l_v, u_v] \subseteq \mathcal{R}$ denotes the domain of the attribute. A set of linguistic terms can be defined over the domain of each quantitative attribute. Let $L_{vT}, r = 1, 2, \dots, \mathfrak{s}$ be linguistic terms associated with some quantitative attribute, $I_v \in J$. L_{vT} is represented by a fuzzy set, L_{vT} , defined on $\text{dom}(I_v)$ whose membership function is \underline{m}_{vT} such that

$$\underline{m}_{vT} : \text{dom}(I_v) \rightarrow [0,1]$$

3.6 Knowledge Discovery through Case Reasoning

A case based reasoner is heavily dependent on the structure and content of its collection of cases, the case library. The representation problem in case based reasoning is primarily the problem of deciding what to store in a case, finding an appropriate structure for describing the content of cases, and deciding how the case memory should be organised and indexed for effective and efficient storage, retrieval and reuse. An additional problem is the integration of the case memory structure with a model of deep, general domain knowledge (inferential, incomplete, uncertain, auxiliary, similarity assessment knowledge) (Lee et al. 1996).

Crucial issues for the representation of case-knowledge involve the *flexibility* of the representation and the *efficiency* of case retrieval. Case-knowledge representation schemes that are too restrictive in terms of what types of

knowledge can be expressed and also how stored knowledge items can be accessed and processed. Furthermore, case based reasoning applications will not be accepted unless case access and the associated reasoning processes operate within certain time limits. Of course, representational flexibility and retrieval efficiency should not be viewed in isolation; generally, the higher the flexibility of representation the lower the efficiency of case retrieval. So a versatile representation regime is one that provides the system designer with a variety of representation elements that allow him or her to find a compromise between flexibility and efficiency for a particular application. Other current issues in case based reasoning that have no direct reflection on the knowledge representation side include *scaling-up*, *evaluation*, *tools*, and *methodologies*.

A learning system should make increasingly useful decisions as it accumulates experience. This is the express goal of the work in CBR, an approach within the explicit methodology. With any approach, CBR has its advantages and limitations. The CBR methodology can be effective even if the knowledge base is imperfect. Certain techniques of automated learning, such as explanation-based learning, work well only if a strong domain theory exists. In contrast, case reasoning can use many examples to overcome the gaps in a weak domain theory while still taking advantage of the domain theory. CBR can also be used when the descriptions of the cases, as well as the domain theory, are incomplete. A further advantage of CBR is the relative ease of combining techniques with other approaches such as production rules (Adomavicius and Tuzbilian 2001). An example of such compatibility is a system which uses case reasoning to solve problems whenever possible; otherwise it resorts to heuristics to decompose a problem into a simpler one.

In general, the prior cases retrieved by case based reasoning will match the required solution only imperfectly. In particular, the source cases may fail to fulfill some of the requisite objectives. At this point, an analogy can be formed between the functionality of the precedent solutions and the goals of the current problem. The prior solutions may then be modified to eliminate or circumvent the limitations. Then a process of iterative refinement can be employed to adapt an old solution to the new problem context. Whether or not analogy is used, an organization may be imposed on the case base through the use of clustering technique (Kim and Novick 1993). In this way, a target case may be readily accommodated into an existing case base. In a larger sense, CBR is difficult to avoid in practical contexts. Decisions are made against the light of past experience, whether such information is encoded into a machine-compatible format

or is available only informally. In addition, each new problem and the attendant solution constitute a case.

3.7 An Integrated Methodology for Learning

Case-based reasoning and artificial neural networks are complementary problem solving methods (Lees and Corchado 1998). CBR has the potential to provide, by reference to previous learned experiences, problem solving capabilities in situations which defy attempts to obtain a satisfactory solution through the use of logical, analytical techniques of knowledge-based systems when a clear model of the problem domain is unobtainable. Neural networks are able to analyze large quantities of data to establish general patterns and characteristics in situations where rules are not known and, in many cases, can make sense of incomplete or noisy data. Furthermore, while neural networks deal easily, and normally, with numeric data sets, case-based reasoning can also handle more complex symbolic knowledge structures (Malek 2001).

CBR as a problem-solving paradigm assumes the minimum of two processes: recall and adaptation. These areas have the potential for integrating the concept of CBR with other AI techniques (Maher 1998).

1. Acquiring memory and memory indices: AI techniques such as knowledge acquisition techniques, and machine learning techniques with conceptual clustering are useful.
2. Recalling cases: machine learning such as induction can be used to develop an indexing tree, and pattern matching and similarity measures can be used for selecting cases from case memory.
3. Adaptation: various problem solving paradigms such as constraint satisfaction, satiation heuristic search, genetic algorithms can be used depending on what kind of knowledge is available.

The architecture of our integrated model is shown in Figure 2.

There are 3 stages of processing, namely, CBR case profiles building process, separated data training by means of NN and CBR and result evaluation from above training output. The first stage is data processing to discover rules from the data. By incorporating with expert knowledge, only those valid rules are accepted during the validation process. Hence various case profiles are established with reference to individual valid rule. Two data sets are constructed in terms of machine reliability and station patronage level. In the second stage, these two data sets are used as inputs for the network training through NN and CBR. In the final stage, outputs from NN and CBR are evaluated with an aim to identify individual optimization.

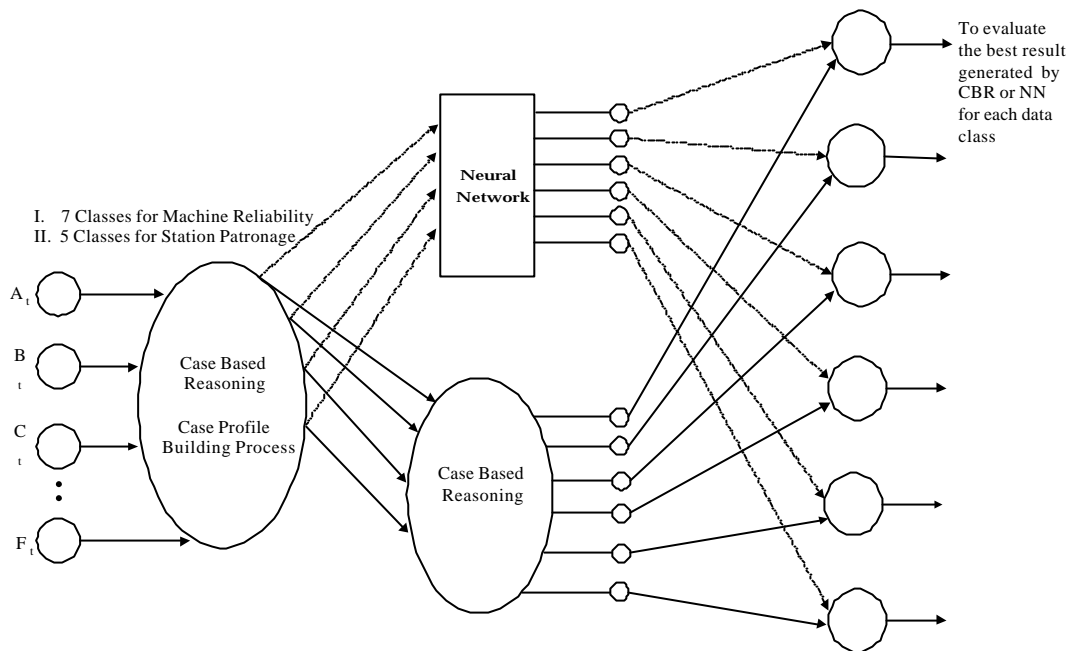


Figure 2 Architecture for the integrated methodology

The purpose is to identify which particular situations and conditions that this integrated approach provides the best result when combining CBR and evolution strategy. This optimization process can even construct a practical hybrid industrial application. Building case profiles (Adomavicius and Tuzbilian 2001) from the database, as shown in Figure 3, it consists of two main phases of CBR case profile building process: rule discovery and validation. In rule discovery phase, each record is modeled with various types of conjunctive rules, including association and classification rules. By incorporating expert knowledge,

only those valid rules are accepted during the validation process. Since CBR generates large numbers of rules, one way to validate discovered rules is to let a domain expert study and decide how well they represent the actual condition. The expert accepts some rules and rejects the rest and those undecided rules. Fine-tuning and re-validation processes are required in order to include some extreme scenarios. All accepted rules form the cases of machine performance profiles in our domain of interest.

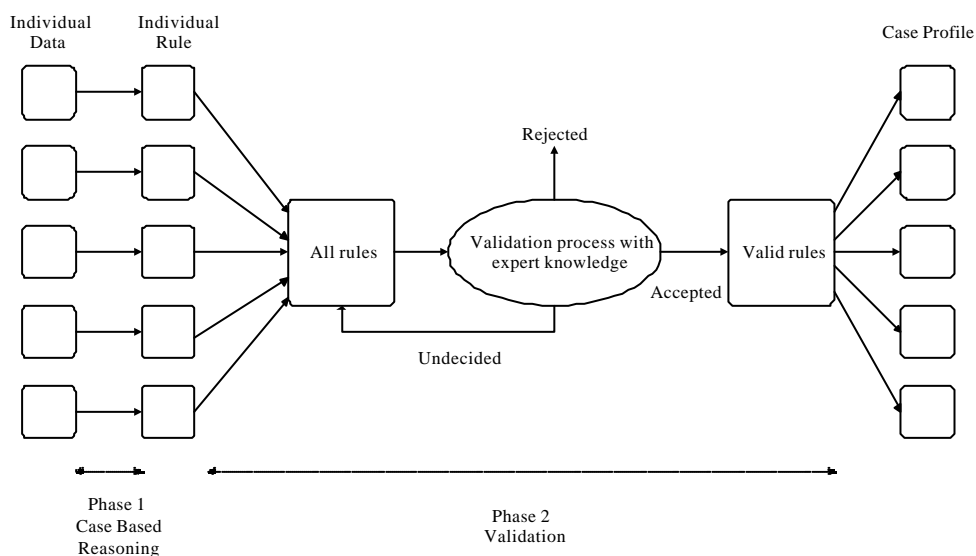


Figure 3 CBR case profiles building process

Two data sets are constructed in terms of machine reliability and station patronage level for this experiment. Practically, the domain expert applies two types of validation techniques:

1) *Similarity-based grouping*

This puts similar rules into groups according to expert-specified similarity criteria. The expert can inspect groups of rules instead of individual rules one by one, and can accept or reject all rules in the group at once. In the data set of station patronage, those data with similar condition of station busy level in term of daily patronage transactions are grouped together into five classes as comparing with machine daily transactions.

2) *Template-based filtering*

This step filters rules that match expert-specified rule templates. The expert specifies accepting and rejecting templates. As a result, rules that match accepting template are accepted to form a set of valid rules. Rules that match a rejecting template are rejected. Rules that do not match a template remain invalidated and re-cycle the process if necessary. In the data set of machine reliability, seven templates are designed to filter data with reference to the cumulated machine usage transactions as comparing with maintenance scheduling pattern since their last maintenance history. Those data with extremely low transactions are rejected in an exemption log for further verification.

4. Experimental Results

The same database being studied in (Liu and Sin 1997, 1999, 2000) was utilized for the simulations as the illustration on optimization using the integration of case based reasoning and neural network. For data pre-processing, we classify the data by using Visual Basic Language for building the CBR case profiles as shown in Figure 3. Two techniques have been applied: similarity-based grouping and template-based filtering for rule discovery through data analysis. Firstly, we apply similarity rules into groups according to expert-specified similarity criteria. In terms of station patronage, five classes are categorized with the case profiles in the perspectives of station patronage level. Secondly, the template is used to filter accepting rules. For the data set of machine reliability, those machines with similarity in cumulated machine usage transactions are filtered into seven classes as comparing with maintenance scheduling pattern since their last maintenance history. We find out the classification principle from a pre-classified training data. One attribute is seen as a target attribute for the output classification and other attributes are seen as input data pattern.

For example, according to the attribute structure similarity condition of station busy level in terms of daily patronage transactions, all data are classified into five groups. Each group has a range of 5 to 11 stations, while each station has various quantities of AFC machines. Each machine has individual characteristic performance with reference to geographical location of each station, machine type, daily machine usage transactions, number of ticket rejections, maintenance scheduling pattern and cumulated machine usage transactions since last maintenance activity. On the other hand, we filter data that match expert-specified rule template. For expert knowledge, seven types of filtering criteria on machine reliability in term of failure rate are developed. The machine reliability is defined as the machine failure condition since last maintenance activity as well as the effectiveness of individual maintenance cycle.

After the generation of CBR case profiles, we arrange above two types of pre-processed data sets from similarity-based grouping and template-based filtering separately into the neural network training and case-based reasoning through *NeuralSIM*¹ with our integrated model as shown in Figure 2. During the training process, we discover that the rate of converge is not satisfactory and we process the data again by applying fuzzy clustering with an aim to discover and fine-tune any useful classes of items. For data set of machine reliability, 7 classes are defined with various from 6% to 38% of the population, thus representing a range of 158 to 1,191 cases in each class. Similarly, for data set of station patronage, 5 classes are set-up with 9% to 25.5% of the population and each class has a data range of 276 to 776 cases.

As the data being categorized into two main perspectives of machine reliability and station patronage, their results are summarized in Tables 1 and 2 respectively. The adaptive gradient learning algorithm with various network architectures for machine reliability are set-up with ranges between 11-10-1 to 6-10-1, while the network for station patronage are built with ranges between 8-2-1 to 4-4-1. By comparing outputs from CBR and NN, results are evaluated in order to identify individual optimization. Tables 1 and 2 also indicate relative results with corresponding percentage of improvements for each data class of machine reliability and station patronage.

From the experimental results of integrated approach, we notice that CBR training produces best results for some particular data classes, while NN works best for the rest as highlighted in Tables 1 and 2. That is, CBR generates best results for data classes 2, 5, 6 and 7 of machine reliability and data classes 1 and 5 of station patronage. As an

¹ NeuralSIM product of Aspen Technology, Inc., 202 Park West Drive, Pittsburgh, PA 15275.

effective measure of our proposed model against existing maintenance practice, a simulated maintenance arrangement using existing approach is configured to highlight the degree of improvement. The improvements of our proposed model in terms of machine reliability and station patronage are summarized in Tables 1 and 2. The improvement ranges are varied between 26% to 7,011% and 300% to 1,246% in machine reliability and station patronage respectively. Upon the experimental results of our proposed model, a productivity enhancement suggestion on machine maintenance scheduling concludes that savings in terms of labor, materials and other overheads could be achieved significantly.

The main objective of this experiment is to identify an enhanced optimization approach by integrating data clustering technique as well as learning algorithm. System improvement can even further achieve by applying different learning parameters of CBR and NN. By comparing previous training results in (Liu and Sin 1997, 1999, 2000), it illustrates the importance of categorizing case profiles and integrating data clustering technique before network training. Finally, it is proved that CBR or NN works well against each other according to individual characteristics as performed from different data classes. This provides some insights for future researches.

Machine Reliability			Algorithms	Experimental Error Result from Machine Reliability Perspective		
Class	Cumulated Transactions	% of Population		Proposed Model	Simulated Arrangement using Existing Practice	Improvement (%)
1	Below 50K	38%	NN	0.0115	0.0798	594%
			CBR	0.0342	0.2434	612%
2	50 - 100K	19%	NN	0.0064	0.0268	317%
			CBR	0.0032	0.0225	811%
3	101 - 150K	13.5%	NN	0.0056	0.0211	275%
			CBR	0.0062	0.0170	174%
4	150 - 200K	9%	NN	0.0050	0.0106	113%
			CBR	0.0079	0.0055	30%
5	201 - 250K	6%	NN	0.0082	0.0220	169%
			CBR	0.0029	0.0037	26%
6	251 - 300K	5.5%	NN	0.0053	0.0135	154%
			CBR	0.0053	0.0121	367%
7	Above 300K	9%	NN	0.0406	0.1281	216%
			CBR	0.0032	0.2241	7,011%

Table 1 Preliminary training result from Machine Reliability Perspective

Station Patronage			Algorithms	Experimental Error Result from Station Patronage Perspective		
Class	Transaction Range	% of Population		Proposed Model	Simulated Arrangement using Existing Practice	Improvement (%)
1	Low	9%	NN	0.0767	0.2542	431%
			CBR	0.0745	0.4741	737%
2	Medium	21%	NN	0.1051	0.4040	485%
			CBR	0.1963	1.1402	681%
3	High	23.5%	NN	0.1436	0.2874	300%
			CBR	0.2524	0.0631	350%
4	Very high	25.5%	NN	0.1192	0.3593	411%
			CBR	0.1668	0.0055	732%
5	Extremely high	21%	NN	0.1193	0.4055	441%
			CBR	0.1146	1.1365	1,246%

Table 2 Preliminary training result from Station Patronage Perspective

5. Conclusion and Discussion

The aim of present research is to construct an intelligent system model with an enhanced optimization approach by integrating data clustering technique as well as learning algorithms of CBR and NN. Machine maintenance scheduling is used for an illustrated purpose. From the points of reliable machine performance and manpower productivity perspectives, it is a challenge to adopt a system that can optimize our limited resources in the modern competitive environment. This drives the need to establish optimizing methodology and mechanism to facilitate a smooth and easier daily activities as a champion among the current market. Data analysis is performed basically to retrieve and discover various rules, profiles and knowledge from the database. A data clustering model is essential for effective data categorization before applying learning algorithm. Genetic Algorithms for evaluating the orderings and generating recommendations are encountered. This hybrid optimization approach will provide a system for various applications with specific domain of interest.

From our research result, it provides an insight that each data class level of machine reliability and station patronage generates different degree of improvements against each other. This integrated model is evaluated as the most effective maintenance scheduling arrangement as shown in Table 3. As comparing with previous studies, there are improvements with 9,113%, 2,341%, 1,127% and 246% on maintenance scheduling against other algorithms through FastPropTanh (FT) (Liu and Sin 1997), CBR, TLRN (Liu and Sin 1999), and combining CBR with TLRN (Liu and Sin 2000) respectively. The current study provides the best result by integrating data clustering technique as well as learning algorithms of CBR and NN. This indicates the need to develop an intelligent hybrid system for optimization purpose. Furthermore, learning capabilities in both CBR and ES will still be a subject of future research. It is anticipated to apply the case adaptation technique by chromosome replacement logic between CBR and ES. This enables further enhancement on training parameters for achieving better results. The purpose is to construct a prototype with efficient and practical hybrid approach for optimizing various determining factors in the domain of interest.

Method	Previous study results through various algorithms [21, 22, 23]				Present research
Network	Fast Prop Tanh (FT)	CBR	Time Lagged Re-current Network (TLRN)	CBR + TLRN	CBR / NN
Result	0.2635	0.0698	0.0351	0.0099	0.0029
Improvement (%)	9,113	2,341	1127	246	Best result

Table 3 Results Comparison with Previous Studies

Acknowledgement

The authors appreciate the partial support from the Central Research Grants G-V579 of The Hong Kong Polytechnic University.

References

- [1] Aamodt, A., and Plaza, E. 1994. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AI Communications*. IOS Press 7(1):39-59.
- [2] Adomavicius, G., and Tuzbilian, A. 2001. Using Data Mining Methods to Build Customer Profiles. *Computer – Innovative Technology for Computer Professionals*. Feb., 2001 :74-82.
- [3] Aha, David W. 1998. The Omnipresence of Case-based Reasoning in Science and Application. *Knowledge-based Systems* 11(5-6):261-273.
- [4] Aha, David W.; Breslow L.A.; and Munoz-Avila, H. 2000. Conversational Case-based Reasoning. *Applied Intelligence* 2000: 1-25.
- [5] Au W. H., and Chan, Keith C.C. 1999. FARM: A Data Mining System for Discovering Fuzzy Association Rules. *1999 IEEE International Fuzzy Systems Conference Proceedings* 3:1217-1222.
- [6] Azuaje, F.; Dubitzky, W.; Black, N.; and Adamson, K. 2000. A Review on Retrieval Strategies for Case Based Reasoning. In *Knowledge Engineering Review* 2000.
- [7] Back, T.; Hoffmeister, F.; and Schwefel, H.-P. 1991. A Survey of Evolution Strategies. *Proceedings of the Fourth International Conference on Genetic*

Algorithms, Morgan Kaufmann Publishers, Los Altos, CA.: 2-9.

- [8] Baluja, S. 1994. Population-based incremental learning : A method for integrating genetic search based function optimization and competitive learning. *Tech Rep. CMU-CS-94-164, Dept. Comp. Sci., Carnegie-mellon Uni.*
- [9] Burkhard, H.-D., and Richter, M. 2001. On the Notion of Similarity in Case Based Reasoning and Fuzzy Theory. In *Soft Computing and Case based Reasoning*. Springer-Verlag, Pal, S. K. (ed.), Dillon, T. S. (ed.), Yeung, D. S. (ed.):29-45.
- [10] Dubitzky, W., and Azuaje, F. 2001. A Genetic Algorithm and Growing Cell Structure Approach to Learning Case Retrieval Structures. In *Soft Computing and Case based Reasoning*. Springer-Verlag, Pal, S. K. (ed.), Dillon, T. S. (ed.), Yeung, D. S. (ed.):115-146.
- [11] Golding A. R., and Rosenbloom P. S. 1991. Improving rule-based systems through case-based reasoning. *Proceedings of AAAI-91* 1 :22-27.
- [12] Han, J.; Huang, Y.; Cercone, N.; and Fu, Y. 1996. Intelligent Query Answering by Knowledge Discovery Techniques. *IEEE Transactions on Knowledge and Data Engineering* 8(3):373-389.
- [13] Ishii, N., and Wang, Y. 1996. Learning Feature Weights for Similarity Measures using Genetic Algorithms. *Journal of Japanese Society for AI* 1996.
- [14] Job, D.; Shankararaman, V.; and Miller, J. 1999. Combining CBR and GA for Designing FPGAs. *IEEE* 1999 :133-137.
- [15] Kim S. H., and Novick M. B. 1993. Using clustering techniques support case reasoning. *International Journal of Computing Applications in Technology* 6 :57-73.
- [16] Lee, K. M.; Kwak, D. H.; and Hyung, L. K. 1996. Fuzzy Inference Neural Network for Fuzzy Model Tuning. *IEEE Transactions on Systems, Man, and Cybernetics* 26(4):637-645.
- [17] Lees, B., and Corchado J. M. 1998. Neural Network Support in a Hybrid Case-Based Forecasting System. In *Case-Based Reasoning Integrations*, Technical Report WS-98-15, AAAI Press: 85-90.
- [18] Liu, James N. K. 1997. Quality Prediction for Concrete Manufacturing. *Automation in Construction*. 183, Elsevier Science.
- [19] Liu, James N. K., and Sin, Danny K. Y. 1997. Fuzzy Neural Network for Machine Maintenance in Mass Transit Railway System. *IEEE Transactions on Neural Networks* 8(4).
- [20] Liu, James N. K., and Sin, Danny K. Y. 1999. Evaluating Case-based Reasoning and Evolution Strategies for Machine Maintenance. *1999 IEEE International Conference on Systems, Man, and Cybernetics II* :480-485.
- [21] Liu, James N. K., and Sin, Danny K. Y. 2000. An Integrated Data Mining Approach for Maintenance Scheduling. *Engineering Intelligent Systems* 8(2):119-126.
- [22] Maher, M. L. 1998. CBR as a Framework for Design: Augmenting CBR with other AI techniques. *Case-Based Reasoning Integrations*. Technical Report WS-98-15, AAAI Press :96-101.
- [23] Malek M. 2001. Hybrid Approach for Integrating Neural Networks and Case Based Reasoning: From Loosely Coupled to Tightly Coupled Models. In *Soft Computing and Case based Reasoning*, Springer-Verlag, Pal, S. K. (ed.), Dillon, T. S. (ed.), Yeung, D. S. (ed.):73 -94.
- [24] Munoz-Avila, H.; Hendler, J.A.; and Aha, David W. 1999. Conversational Case-based Planning. *Review of Applied Expert Systems* 5 :163-174.
- [25] Plaza, E.; Arcos, J. L.; and Franciso, and Nartin. 1997. Cooperative Case-based Reasoning. *Lecture Notes in Artificial Intelligence*. Springer Verlag, 1221:180-201.
- [26] Quagliarella, D.; Periaux, J.; Poloni, C.; and Winter, G. 1998. Genetic Algorithms and Evolution Strategy in Engineering and Computer Science. Wiley.
- [27] Rajat, K. De., and Pal, S. K. 2000. Pattern Classification using Fuzzy Sets and Neural Nets: a Case-based Approach. *Engineering Intelligent Systems* 8(2):103-108.
- [28] Smyth, B. 1999. Constructing Competent Case based Reasoners: Theories, Tools and Techniques. *Proceedings of the Workshop on Automating the Construction of Case Based Reasoners, IJCAI-99, 1999, Stockholm, Sweden*.
- [29] Watson, I., and Marir, F. 1994. Case-Based Reasoning: A Review. *Knowledge Engineering Review* 9(4) 1994.